AMENDMENTS TO THE SPECIFICATION:

Replace the Abstract with the following:

A method (and structure) of improving at least one of speed and efficiency when executing a linear algebra subroutine on a computer having a memory hierarchical structure including at least one cache, the computer having M levels of caches and a main memory. Based on sizes, it is determined, for a level 3 matrix multiplication processing, which matrix will have data for a submatrix block residing in a lower level cache of the computer and which two matrices will have data for submatrix blocks residing in at least one higher level cache or a memory. From a plurality of six kernels, two kernels are selected as optimal to use for executing the level 3 matrix multiplication processing as data streams from different levels of the M levels of cache, such that the processor will switch back and forth between the two selected kernels as streaming data traverses the different levels of cache. Data from the selected two matrices is streamed, for executing the level 3 matrix multiplication processing, so that the submatrix block residing in the lower level cache remains resident in the lower level cache.

Replace the text beginning at line 1 of page 5 through line 9 of page 6 by the following two paragraphs:

To achieve the above and other exemplary purposes, in a first exemplary aspect of the present invention, described herein is a method of improving at least one of speed and efficiency when executing a linear algebra subroutine on a computer having a memory hierarchical structure including at least one cache, the computer having M levels of caches and a main memory. The method includes determining, based on sizes, for a level 3 matrix multiplication processing, which matrix will have data for a submatrix block residing in a lower level cache of the computer and which two matrices will have data for submatrix blocks residing in at least one higher level cache or a memory. Two kernels, from a plurality of six kernels, are selected as optimal to use for executing the level 3 matrix multiplication processing as data streams from different levels of the M levels of cache, such that the processor will switch back and forth between the two selected kernels as streaming data traverses the different levels of cache. Data is streamed from the selected two matrices, for executing the level 3 matrix multiplication processing, so that the submatrix block residing in the lower level cache remains resident in the lower level cache.

In a second exemplary aspect of the present invention, also described herein is an apparatus including a memory system to store matrix data for a level 3 matrix multiplication processing using data from a first matrix, a second matrix, and a third matrix, the memory system including at least one cache, and a processor to perform the level 3 matrix multiplication processing. Data from one of the first matrix, the second matrix, and the third matrix is stored as a submatrix block resident in a lower level cache in a matrix format and data from a remaining two matrices is stored as submatrix blocks in the memory system at a level in the memory system higher than the lower level cache. The processor preliminarily

selects, based on sizes, which matrix will have the submatrix block stored in the lower level cache and which two matrices will have submatrix blocks stored in the higher level. Data from the selected two matrices is streamed through the lower level cache into the processor, as required by the level 3 matrix multiplication processing, so that the submatrix block stored in the lower level cache remains resident in the lower level cache. The computer has M levels of caches and a main memory, and the processor further preliminarily selects, from a plurality of six kernels, two kernels optimal to use for executing the level 3 matrix multiplication processing as data streams from different levels of the M levels of cache, such that the processor switches back and forth between the two selected kernels as streaming data traverses the different levels of cache.